Association for Information Systems
AIS Electronic Library (AISeL)

ECIS 2008 Proceedings

European Conference on Information Systems (ECIS)

2008

# E3-Service - A Model-Based Approach for Generating Needs-Driven E-Service Bundles in a Networked Enterprise

Sybren De Kinderen
*VU University*, sdkinde@few.vu.nl

Jaap Gordijn
*VU University Amsterdam*, gordijn@few.vu.nl

# $E^3$SERVICE: A MODEL-BASED APPROACH FOR GENERATING NEEDS-DRIVEN E-SERVICE BUNDLES IN A NETWORKED ENTERPRISE

De Kinderen, Sybren, VU University, de Boelelaan 1081a, 1081HV Amsterdam, The Netherlands, sdkinde@few.vu.nl

Gordijn, Jaap, VU University, de Boelelaan 1081a, 1081HV Amsterdam, The Netherlands, gordijn@few.vu.nl

## Abstract

*e-services are just like normal services, but can be ordered and provisioned via the Internet completely. Increasingly, these e-services are offered as a multi-supplier bundle of elementary services. How to automatically compose these e-service bundles is considered as a key problem. Part of the composition process is to elicit the customer need for an e-service bundle, and to facilitate this elicitation process by a web-enabled software component. To this end, we need a computer-processable theory, called an ontology, which is able to represent customer needs, and also facilitates computer-supported elicitation of needs. In this paper, we propose the* $e^3$service *ontology that does just that. The ontology represents needs, wants, demands, benefits, and related constructs, to build a customer-oriented catalogue of customer needs for e-services, which in turn can be used during the need-elicitation process for a specific customer. We show how the* $e^3$service *ontology works in practice, by using a case study carried out in the field about postal services.*

*Keywords: E-services, service bundling, model-based approach.*

# 1 INTRODUCTION

In recent years, *customizable* e-service bundles, satisfying specific customer needs, have gained interest. e-Services are just like normal commercial services, but can also be ordered and provisioned via the Internet. For instance, consider the commodity internet services, as obtained from an Internet Service Provider (ISP). Usually, an ISP offers a standard, fits-for-all, e-service bundle, eg. consisting of an IP-based access service, an email box service, website hosting, an IP-telephony service, and access to newsgroups. However, the individual customer perhaps prefers a smaller bundle; e.g. *only* IP-based access plus email. Such a custom-made bundle then more closely matches the customer need compared to the fits for all, full-service bundle.

In addition to the customization of e-service bundles, e-services are increasingly offered by a *networked* value constellation, rather than a *single* enterprise (Tapscott, 2000). By doing so, suppliers can utilize their core competencies, while still satisfying a complex customer need. Returning to the ISP-example, an e-service bundle can indeed be a *multi-supplier* bundle: IP-access is then provided by a telecom operator, an email box is offered by a commercial enterprise utilizing economies of scale, as can hold for website hosting, which may be offered by yet another enterprise. Jointly, these enterprises satisfy one complex need.

We perceive the issue of how to *automatically* compose and provision these *multi-supplier* e-*service bundles* as a key research problem. We envision a scenario in which a customer states his need to the web, and the web responds with a list of candidate multi-supplier e-service bundles, satisfying the stated need. Then, after selection of the preferred bundle by the customer, the e-services in the bundle should be provisioned automatically. To enable computer supported customer *need elicitation*, and to *match* found needs with available e-services in the market, a machine processable theory about customer needs and multi-supplier e-services is required. Such a theory represents needs and e-services, and reasons about need-service matching and bundling. In this paper, we focus on need-elicitation; for matching needs and services see Baida (2006).

Guidelines on how to create customized service bundles based on needs have already been studied in service marketing, most notably by Grönroos (1990) and Lovelock (2001). However, these guidelines lack conceptualization and formalization, meaning that it is not possible to (semi-) automatically, and so computationally, reason about service bundles yet.

The contribution of this paper is a stepwise approach called *e³service*, which allows for semi-automated customer need elicitation, and matching of needs with available e-services, based on earlier found e-service catalogues, and need types. To do so, the *e³service* approach formalizes well-known concepts found in service marketing. We understand e-services as *commercial services*: economic activities, deeds and performances of a mostly intangible nature (Normann, 2000), with a focus on those services that can be ordered and provisioned (nearly) online. This is in contrast with *web services* and related standards such as BPEL4WS (2003): these services are mainly intended to arrive at a cross-organizational computing platform to facilitate interoperability on a more *technical* level.

This paper is structured as follows. In section 2, we present the *e³service* conceptual model that enables formally modelling services from a customer need perspective. Section 3 then provides a high-level overview of the entire *e³service* approach, to explain how we generate bundles of services and how the *e³service* conceptual model fits in. In section 4 we apply this ontology to a real-life case study performed in the Dutch postal industry, to reason about potential service bundles. In section 5, we discuss related work on e-service bundling. Finally, in section 6 we present our conclusions.

# 2 *THE E³SERVICE* ONTOLOGY

To facilitate automated reasoning about service bundling, we utilize established service marketing literature (see Grönroos (1990), Lovelock(2001) and Normann(2000)). We express key service marketing concepts as a formal ontology, which is an explicit, formal, and shared conceptualization of a domain (here: e-services) (Borst,1997). An ontology is first a *formal* specification, enabling automatic reasoning about needs elicitation, need-service matching and service bundling. Second, an ontology is a *shared* conceptualization meaning that stakeholders share the semantics of concepts in the ontology. Since we want to reason about needs and service bundling in a *networked* enterprise, it is important that each organization in this network understands concepts (needs and services) in the same way to avoid mismatches. To reach this shared understanding, the ontology must be based on agreed knowledge. That is why we base our ontology on *established theory* from service marketing literature.

Figure 1 presents the *e³service* ontology. An elaborate version can be found in (de Kinderen, 2007). In this paper, we only give a concise summary with the purpose to make the paper self-contained. This summary is organized by identifying the most important parts in the ontology (1) the need/demand/want hierarchy, (2) benefits and consequences (3) services, and (4) dependencies between want/consequence pairs.

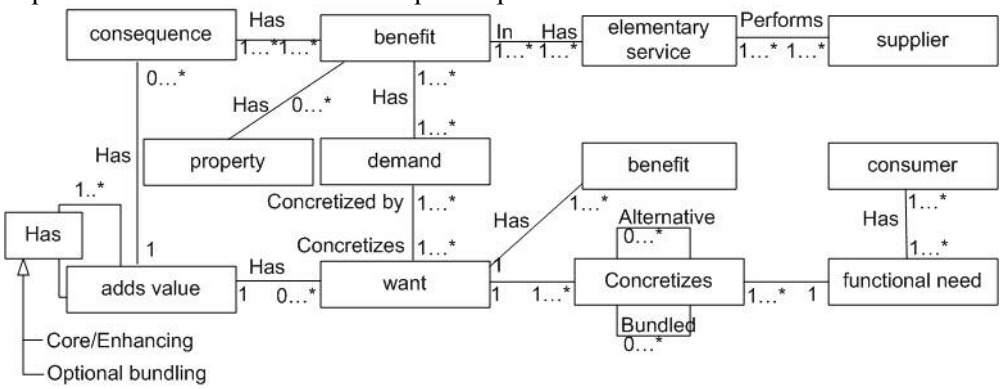


*Figure 1: The* e³service *ontology formalizing customer needs, cf. UML 2.0*

## 2.1 The need/want/demand-hierarchy

The need/want/demand-hierarchy emphasizes a *gradual transition* from a need - a problem statement - to a set of services that together provide a solution for that need, or a demand. (see Arndt (1978) and Kotler ( 2000)). For the *e³service* ontology, it results in the following concepts.

**Functional need.** A functional need represents a problem statement or goal, independently of a *solution direction*. A need (problem) can usually be covered by multiple *alternative* wants, and ultimately demands (solution) (Arndt, 1978). Also, a need may require multiple *partial* wants for satisfaction. The separation of problem and solutions is important to avoid that we overlook alterative wants (solutions) for needs (problems) during the elicitation process.

*EXAMPLE:* 'communicating with family abroad'. Note that this need statement does *not* include a notion of a solution yet as nothing is stated about *how* the communication will be done.

**Want.** A want specifies an *initial* solution direction for a need. The want is an initial solution direction, because a want does not indicate a specific (named) supplier satisfying the stated need yet; it indicates a general service that can supplied by any interested supplier. Moreover, a want is something that can be offered by a *single supplier*, if this is commercially feasible. These single suppliers can then jointly satisfy a more complex need, by combining the wants they can satisfy.

*RELATIONS:*

- *Has* benefit: A want has one or more benefits, which are value-properties of a provided service. In case of a want, these properties are defined independently from a specific supplier.

*EXAMPLE:* A want satisfying the need 'communicating with family abroad' is 'e-mail hosting'. A benefit for 'e-mail hosting' is a certain mail-box size, where the actual size of the mail box is unspecified since it depends on the supplier. An alternative want is 'instant messaging'.

**Concretizes:** A want *concretizes* a functional need by specifying an *initial solution* direction. There exist two distinct ways for concretizing a need into multiple wants:

- *Alternative*: Each want *fully* satisfies a single need, but alternative wants exist.
- *Bundled*: Each want *partially* satisfies a need, so we require a bundle of wants to satisfy a *single* need.

*EXAMPLE:* 'e-mail hosting' and 'instant messaging' are alternative wants concretizing the need 'communicating with family abroad'. 'E-mail hosting' plus 'IP-connectivity' exemplify bundled wants, because only this combination satisfies the need 'communicating with family abroad'.

**Demand.** A 'want' is provisioned by a *specific supplier* as a *demand*. A demand differs from a want, as a demand provides *supplier-specific* values to the properties for benefits of a want. We use a distinction between wants and demands, because they refer to two steps in the automated reasoning process about need elicitation and bundling. In the first step, we reason about the required benefits, as contained by a want, to satisfy a need, *independently* from a supplier. In the second step, we reason about the *specific* suppliers who can concretize a want in terms of a demand with specific benefits. This simplifies the reasoning process as the customer first focuses on choosing the benefits (e.g. a mailbox with a 'size' benefit) without a supplier in mind, and thereafter chooses properties for these benefits offered by a supplier (eg. a 1MB sized box).

*RELATIONS: Concretizes*: A demand concretizes a want if it specifies the generic want, for instance e-mail hosting, for a specific supplier. A want has generally one or more demands, meaning that one or more suppliers can satisfy a want.

*EXAMPLE:* 'Gmail' (from Google) is a demand that specifies the want 'e-mail hosting'. For example, 'Gmail' may have a distinguishing property 'mail-box size=1 GB' that would be different from the 'mail-box size=0.5 GB' as offered by 'Hotmail'.

### 2.2 Benefits, consequences and value derivations

**Benefit**. Benefits describe properties that are of economic value to the customer in the sense of *value-in-use* (Ramsay, 2005). In other words, benefits provide an increase of economic utility to the customer, through something functional, social (e.g. status) or otherwise. A benefit is also used to connect demands *as needed by the customer* to services *as provided by the supplier*. Often, there is a mismatch between the set of benefits as contained by a *customer* demand, and the set of benefits as contained by a *supplied* service. In our work, we assume that the customer and the supplier use the *same* terminology to represent a customer/supplier benefit, so ontologically, these benefits are the same. Reasoning about *matching* a customer demand with available supplier E-services is then about finding a *multi-supplier IT service bundle* with a set of benefits, that comes closest the required set of benefits *as contained by the customer demand*.

*RELATIONS:* A want (and a demand also) *has* one or more benefits.

- A single want *has* one or more benefits. Benefits of a *want* have no specific values, as benefits exist *independently* of a specific supplier.
- A single demand *has* one or more benefits. Since a demand *is specific* for a supplier, benefits of a demand do have specific values.

- A service *has* one or more benefits. Services, as seen from a supplier have benefits.

*EXAMPLE:* For of an e-mailing service, a benefit is eg. 'customized domain'. Such a domain allows for customizing an e-mail address, so art@vandelay.com can be used rather than art.vandelay@someunchangebledomain.com. This is a benefit because a customized email address gives the customer more status, heightened stature being a measure of more value in use.

**Consequence.** This concept represents the result in terms of *subjective added value* that the end-customer obtains, if s/he consumes a benefit contained in a service. In the reasoning process, deriving consequences from benefits is based upon the laddering-technique from means-end chaining (Gutman, 1988). In brief, this is done by asking the question 'what happens when we consume service X in which benefit Y is contained? '.

*RELATIONS*: *has:* A benefit *has* one or more consequences. Multiple benefits can point to the same consequence. A consequence indirectly contributes to a need.

*EXAMPLE:* The benefit 'web-based e-mailing access' has the consequence 'cost-effective communication'. 'Cost-effective communication' contributes to satisfying the need 'communicating with family abroad'.

### 2.3 Service

**Service.** A service is of economic value to the end customer, and is provisioned by a supplier. It is the smallest unit that, from a commercial point of view, can be obtained from a supplier. Services are listed in a service catalogue of a supplier. The notion of service allows for connecting the customer-oriented *e³service* ontology to supplier-oriented ontologies (see e.g. (Akkermans & Baida & Gordijn, 2004)).

*EXAMPLE:* 'e-mail hosting'. Note that eg. a mailbox *size* of 1GB is not a service, since the size cannot be provisioned in its own right, but a benefit of the e-mail service it belongs to.

### 2.4 Dependencies between want/consequence pairs

The notion of service-dependencies (see Baida (2006)) indicates that two services depend on each other. For instance, a service can serve as an *option* for another service, or a service may *exclude* meaningful consumption of another service. In (Baida, 2006), this relation exists only from a *supplier* perspective; eg. a paid e-mail service cannot be delivered without a billing service. Such a dependency can also exist from a *customer* perspective; eg. a spam filter adds value for the customer if bundled with an e-mail hosting service.

**Adds value.** As benefits of wants have economic value consequences for the customer, the wants themselves also have consequences. In *e³service*, this is represented as a reified 'adds-value' relationship between one want and one consequence. We have found two kinds of *dependency* relations, which may exist between two or more 'adds value' relations (so between want/consequence pairs):

- *Core/Enhancing*(C/E). A want/consequence pair B provides added value if bundled with a want/consequence pair A. Pair B cannot be acquired independently from A.
- *Optional Bundling* (OB). A want/consequence pair B adds value to a want/consequence A. Yet, in case of an OB relation, A and B can also be acquired separately.

*RELATIONS*:

- An 'adds value' relationship *contains* a single want and a single consequence. This pair represents a commercially feasible offering, plus part of the subjective value gained from consuming a benefit contained within this offering.

- 'Adds value' *has* a relationship with one or more other *adds value* relationships, to represent the actual dependencies. This *has* relationship is a-kind-of core/enhancing or a-kind-of optional bundling relationship.

*EXAMPLE:* The pair 'e-mail ' (want)/'local access to mail' (consequence) is in a Core/Enhancing relationship with pair 'spam-filter' (want)/'reduction in number of unwanted e-mails' (consequence). So, the want 'e-mail' is related to the consequence 'reduction in number of unwanted e-mails' from the want 'spam filter', where the consequence from latter want indicates *why* this relationship exists. Note that a Core/Enhancing relationship is present, because an acquisition of a spam-filter only makes sense in combination with an e-mail service.

# 3 *REASONING WITH E³SERVICE*

Figure 2 explains the overall reasoning process, and makes an explicit distinction between (1) the *creation* of service catalogues (*on beforehand*, and only once), and (2) reasoning *with* these catalogues about feasible *bundles* on a per customer-need basis (for *each stated customer need*).

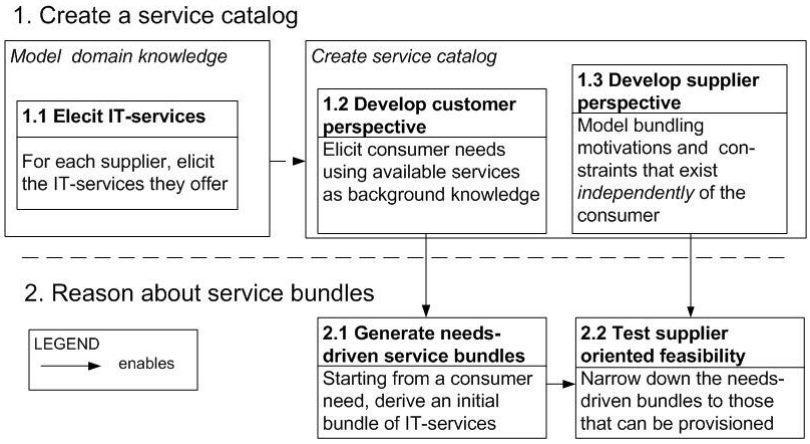


*Figure 2: Steps taken to arrive at a bundle of e-services*

**Step 1.** *Create a service catalogue on beforehand*. *Before* we can actually reason about service bundles themselves, we first have to build per-supplier catalogues that describe the services *and* the needs, wants, demands, benefits and consequences, which can be satisfied by these services. The found catalogues will be used by the automated service bundling reasoning process (see step 2). Building this catalogue requires the following three sub steps:

**Step 1.1.** *Elicit suppliers and for each supplier, elicit the e-services they offer.* We elicit the e-services from those suppliers that like to participate in the bundling process. The result of this task is a -per supplier- list of e-services. These e-services should as fine-grained as possible; nevertheless each service should still be *commercially* feasibly provisioned in its own right. This gives a stop-criterion for decomposing found services into smaller ones.

**Step 1.2.** *Formalize each e-service from a customer perspective.* For each e-service, we elicit and formalize the belonging customer needs, demands, wants, benefits, and consequences, and we state constraints for bundling opportunities by the reified 'adds value' relation.

**Step 1.3.** *Formalize each e-service from a supplier perspective.* For each e-service, we elicit and formalize supplier-related properties to reason about the bundling from the supplier perspective. To do so, we utilize the *serviguration* ontology of (Baida, 2006). Since we focus in this paper on the customer perspective, discussion of the *serviguration* ontology falls outside this paper's scope

**Step 2.** *Generate bundles of services, using the service catalogues*. This step comprises the actual reasoning about matching customer needs with available e-services, and the bundling of these, using the catalogues as found in step 1. We create e-service-bundles satisfying a specific customer need in two steps:

**Step 2.1.** *Derive a set of service bundles that covers a customer need.* We present the customer (alternative) needs, and provide (alternative) wants with benefits. For the selected wants, we present specific demands that can be satisfied by specific suppliers, and use means-end chaining reasoning to analyze value consequences. Finally, we use the 'adds-value' relationship to find want/consequence pairs, and so services that can be added as enhancing or bundled service. This process will exemplified in section 4.

**Step 2.2.** *Narrowing down the set of bundles to feasible bundles from a supply side perspective.* This step utilizes the *serviguration* ontology (Baida, 2006) to find the final set of potential bundles from a supplier perspective. Due to lack of space, we do not elaborate on this step.

We now present steps 1.1, 1.2 and 2.1 in more detail to show how we arrive from a set of individual e-services and a customer need, to a bundle of e-services satisfying that customer need. We focus on the customer perspective only. For supplier perspective (steps 1.3 and 2.2), the *serviguration* ontology (Baida 2006) can be used.

# 4 *E³SERVICE* IN PRACTICE: DIRECT MAIL SERVICES

## 4.1 The Direct Mail case: Enhancing mailings by service bundling

Due to deregulation of the European postal industry, incumbent postal companies operating on the European market have to differentiate themselves in order remain profitable. The Dutch subsidiary of TNT is one such incumbent postal company. In an attempt to differentiate themselves from other postal companies, TNT developed a set of online mailing services that ranges from the ability to design customized stamps online, to an online support service that can aid in creating designs of cards. Amongst these online mailing services, there is an online service that allows a Small to Medium sized Enterprise (SME) to set up a Direct Mail initiative. This service allows a SME to (online) design a mailing and to upload an address list for the mailing recipients. Thereafter, TNT prints the design and physically delivers it to the specified recipients.

Currently the Direct Mail service only allows SME's to send around customized A-5 sized cards, while *additional services* that could enhance the mailing, such as the option of designing a customized stamp (another service of TNT), are never explicitly offered to SME's in combination with the Direct Mail service. In some cases however, the customized A-5 cards do not satisfy the need SME's. Sometimes, to generate sufficient response, the mailing of a specific SME should stand out from others and a single A5-sized card hardly accomplishes this. In order to serve these SME's, TNT has decided that it should provide an option to enhance mailings. TNT would like to accomplish this by bundling the basic Direct Mail service with its other online mailing services, such as the customized stamp mentioned before. Ideally it envisions that an SME goes to TNT's website, states its mailing needs in an online wizard and that, based upon these needs, a bundle of mailing services is presented that matches the mailing needs of the SME. In the following sections, we apply *e³service* to the mailing services of TNT with the purpose of (1) facilitating the creation of such a wizard and (2) to show how the discussed needs ontology works in practice.

## 4.2 Step 1.1. Elicit e-services

We first elicit the individual services for the service catalogue. When eliciting the individual services, the key point in our approach is to consider services from a *value* viewpoint. This means

that we abstract away from the inner workings of a service, such as detailed process descriptions, and instead focus upon the benefits a service provides for the customer. The main reason for considering services from a value viewpoint is that *e³service* is intended for reasoning about how needs can be satisfied by commercial services, and how these services can be meaningfully bundled. Therefore, we leave out detailed descriptions of the inner workings of a service. Through *e³value* modelling (Gordijn 2003), we elicit the following services from TNT: (1) the customized Direct Mail (DM) card (2) customized stamp and (3) DM-advice. We do this elicitation by creating a *e³value* model based upon documentation from TNT. A *e³value* model shows the actors, as well as the commercial services they exchange. Then, the *e³value* model is validated with a domain expert from TNT, who is actively involved in the improvement of TNT's online direct mail service. Additionally, we model mailing services from another supplier: (1) an alternative customized card service and (2) printing additional material such as brochures or vouchers, thereby including multi-supplier characteristics. We choose to take these third party mailing services into consideration, because they provide for a broader coverage of mailing needs. As such, this inclusion increases the likelihood that an SME will actually set up a mailing initiative through the website of TNT. For TNT, this inclusion of third party mailing services can further be advantageous because (1) it can offer its own services in combination with third party services, thus providing an opportunity for sales increase and (2) it can enter into a profit-sharing agreement, in which TNT receives a certain percentage of the income received from each customer it refers. For the third party supplier of mailing services, the profit-sharing agreement is advantageous because it would receive additional customers through TNT. Please note that due to space restrictions, we do not show the *e³value* model in this paper; instead, we directly incorporated mentioned services in the service catalogue depicted in figure 3.

### 4.3 Step 1.2. Create a service catalogue: Develop the customer perspective

**Fill the service catalogue with demands and find the benefits contained within them.** The first step is to fill the service catalogue with the services found by step 1.1. These e-services are actually the demands, since they are the services as provisioned by the specific suppliers. Next, we add the benefits as contained in the demands. Benefits are elicited by searching for specific properties of a service that provide the customer with more value-in-use. For instance, the property 'design template' is a benefit because it saves a SME time when it sets up a mailing initiative. For the demand 'customized card' of TNT, benefits are 'format option', 'paper finish', 'online design letter type' and 'design template'. Note that a demand is not the same as a benefit, since 'format options' cannot be delivered in their own right, whereas a 'customized card' can. The resulting benefits, and the demands that contain them, can be found in figure 3. This figure is actually an instantiation of the *e³service* ontology in figure 1.

**Derive wants.** On the basis of the demands, we derive wants. We first abstract away from the *specific values* that the suppliers give to their benefits of the demand at hand. So, for instance, in the case of a 'format option' benefit for the demand 'customized card', we abstract away from the *supplier-specific* property of providing you with the A3, A4, or A5 formatting options. We create such a supplier-independent property to enable the customer to fill in its preferences, *independently* of having to consider supplier-specific services that can satisfy these preferences

In case multiple, but similar, services available from multiple suppliers, we merge these similar services into one single want. For instance, a demand 'customized card' as provisioned by the specific suppliers TNT and Logiprint, becomes a single want 'customized card', *independently* of these suppliers. This single want will then also inherit the benefits from these different demands. To illustrate, consider the want 'customized card' in figure 3. It contains not only the benefits from TNT's customized card service, but also the benefits from an other supplier, Logiprint.

**Using consequences to show how benefits contribute to satisfying a functional need.** Next, we derive the consequences from the benefits by asking the question: 'What happens when we consume a service in which this benefit is contained? '. For example, take the benefit 'target audience' from our 'mailing addresses business' service, as modelled in figure 3. By specifying the target audience during the consumption of this service, a SME would be able to send their mailing to a specific set of prospects. As such, the consequence of this benefit would be 'enable an SME to send around a mailing more effectively' (compared to choosing prospect addresses at random). By using the consequences, we can now derive a set of needs. We do this in two steps. (1) We first consider for each consequence the *solution-independent* goal that will be achieved by the consequence; these goals then become needs. For example, a consequence 'create a mailing', results in the need 'make an announcement to existing clients'. (2) We cannot always define a need based upon a *single* consequence. Therefore, we also search for groupings of consequences - from *different* wants – to find additional needs. For instance, the grouping of the consequences 'create mailing' and 'reach business prospects' satisfies the need 'attract business prospects'.

**Define dependency relationships between want/consequence pairs.** Now that we have defined both the wants and the consequences, we first search for occurrences of the reified 'adds value'-relationship in *e³service*. Two examples include 'customized card' (want) / 'create mailing' (consequence), and 'direct mail advice' (want) / 'design support'(consequence). Second, we represent dependency relationships between these pairs. For example, the C/E dependency relationship is used here to show that if a 'customized card' is acquired, 'design support' might be something that could also be valuable to the customer. So, if the customer indicates that (s)he is interested in 'design support', the additional want 'direct mail advice' can be offered in combination with a 'customized card'. The result of these steps is that we have now a customer-oriented service catalogue.

### 4.4 Step 1.3: Create a service catalogue: Develop the supplier perspective

In a third step (step 1.3 in figure 2) we develop a catalogue with a supplier perspective on services. See (Baida, 2006) for a detailed discussion of this step.

### 4.5 Step 2.1: Generate needs-driven service bundles

We now illustrate how the service catalogue in figure 3 can be used to generate bundles of services that are tailored to a customer need. To this end, we suppose a scenario in which there are two SME's that consider setting up a mailing initiative: (1) A piano tuner, who is moving and wants to make his new address known to his existing clientele, and (2) A start-up store who wants to create awareness.

**Create an initial bundle of wants.** The piano-tuner starts with a need 'make announcement to existing clients' to create an announcement that he is moving, while the start-up store starts with a need 'attract customer prospects' to create awareness amongst prospect customers. These needs are selected by the piano tuner/start up store from the needs library made in step 1.2. Now, we search in the catalogue for the consequences belonging to these needs. If we consider the piano-tuner, the consequence of the need is 'create mailing'. The exclusive start-up, the selected need has, besides the consequence 'create mailing', also the consequence 'reach customer prospects'. In the service catalogue (figure 3), this combination of consequences satisfies the need 'attract customer prospects' as is indicated by the *AND* annotation. The wants that contain these consequences are the initial bundles *[customized card]* and *[customized card, mailing addresses customer]* for respectively the piano-tuner and the start-up store.

**Involving bundling relationships.** The next step is to expand the found initial bundles with additional wants by considering dependencies between the reified 'adds value' relationships from

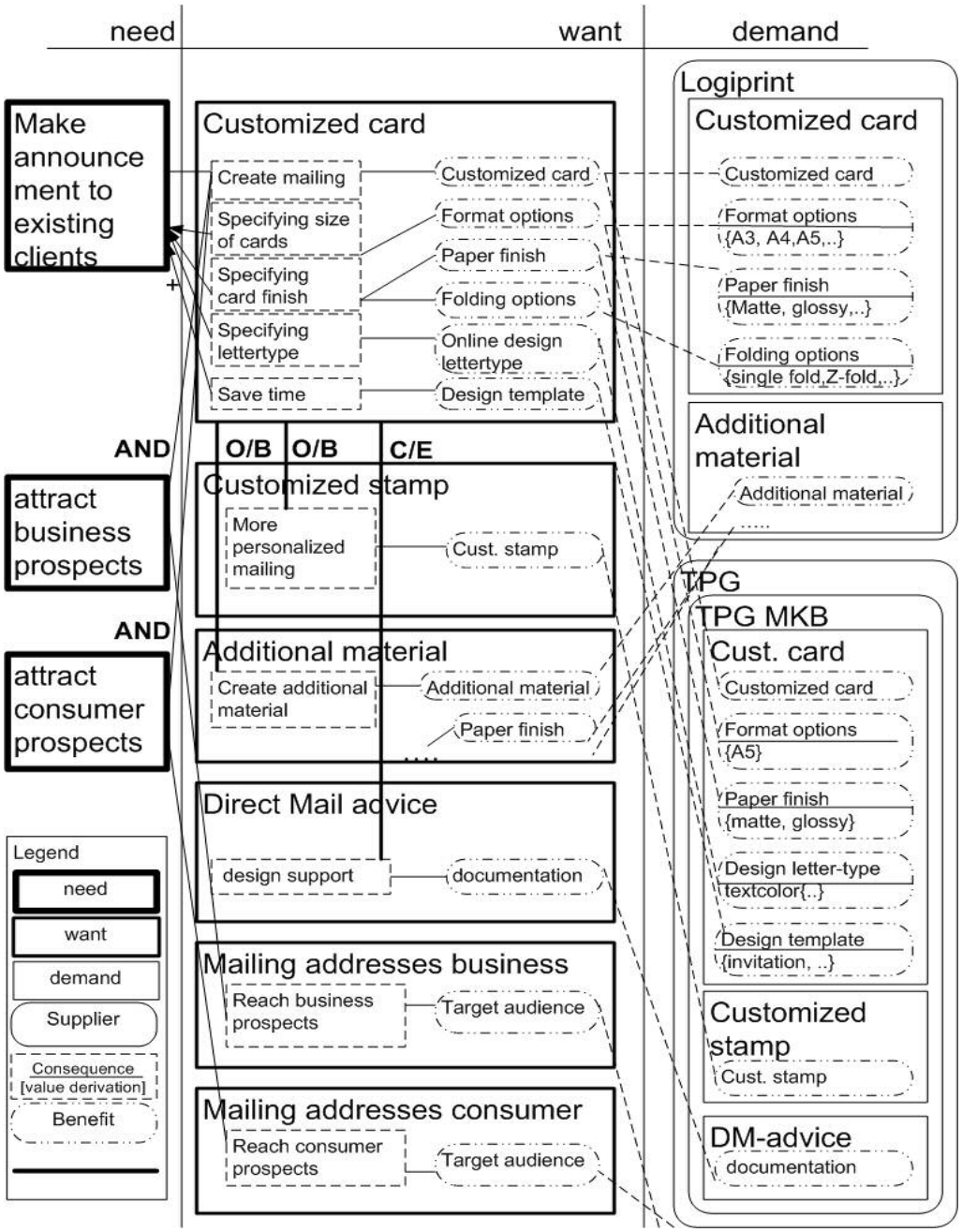


*Figure 3: Customer-driven dependencies between services, cf. the e³service ontology*

the service catalogue. The notion of a consequence is here used by the customer to evaluate whether an additional want should be included in this expansion or not. In other words: we let the customer decide upon inclusion of an additional want in a bundle, based upon the subjective value yielded by consuming that want. For instance, consider the want 'customized card'. According to the dependency relationships that exist in the catalogue, we can expand our initially found pair 'customized card' (want) / 'create mailing'(consequence) with other pairs of wants/consequences. An example of such an additional pair of want/consequence is 'direct mail advice' (want) / 'design support' (consequence), which is in a C/E relationship with 'customized card' / 'create mailing' (see figure 3). Now, we can use the consequences from these additional services to let the customer decide upon inclusion of a particular want in the service bundle.

An exclusive store may have an interest in enhancing his mailing, since it should stand out from other mailings to achieve his main purpose: attracting customers. In this case, we assume that the start-up store is indeed an exclusive store, and so is interested in 'design support' and a 'more personalized mailing'. Since these are the consequences that belong to the wants 'direct mail advice' and 'customized stamp' respectively, we now arrive at the bundle *[customized card, customized stamp, design support, mailing addresses customer]* for the exclusive store. The piano-tuner would not have much use for additional services to enhance his mailing since it mainly serves a practical purpose; informing his existing customers. He will therefore remain at the bundle *[customized card]*.

**Considering the benefits.** Next, we consider the specific benefits that stem from these wants in detail. We need to do so, because we have not yet reviewed all benefits from the wants, such as the available sizes of a customized card. In our example, we assume that the piano-tuner is interested in a customized card for which he can select a design template, since he mainly is interested in getting a message across. The start up store however, wants to use a customized design on a folded, A4 sized card, so that discount-coupons can be attached to their mailing. From this step, we yield the desired benefit 'design template' from the piano-tuner. From the shoe store, we yield the desired benefits 'card size=A4' and 'Folding options=double/triple fold'.

**Relate wants and benefits to demands.** Now that we have found the wants from the SME's and the specific benefits desired, the last step is to select which actual service offerings from specific suppliers provision the services satisfying these wants and benefits. For the piano-tuner, we arrive at the bundle *[customized card(TNT)]*, since TNT can provide him with a design template. Concerning the start up store, we arrive at the bundle *[customized card (logiprint), customized stamp(TNT), design support(TNT), mailing addresses customer (TNT)]*. In case the supplier perspective is involved also, by using the *serviguration* ontology (Baida 2006), additional services may found, e.g. as a result of supply-side bundling analysis.

### 4.6 Step 2.2: narrow down needs-driven bundles to those that can realistically be provisioned

Finally, we narrow down the generated bundles to those that can actually be provisioned from a supplier perspective (step 2.2 in figure 2). See (Baida, 2006) for a detailed discussion of this step.

## 5 RELATED WORK

The *Business Motivation Model* (BMM) (B.R. Group, 2005) is a model representing *ends* (goals, objectives) that are to be achieved by *means*. It abstracts away from implementation issues such as the business processes necessary to provide for the means. In comparison to our work, BMM does not explicitly assist in deriving customer needs from a set of e-services. Also, it does not take a multi-supplier perspective. *Serviguration* (service configuration) (Baida, 2006) provides computer supported reasoning about general service bundles. Case studies in the realm of electricity supply and healthcare have shown that by using this ontology, meaningful bundles of services can generated semi-automatically (Akkermans & Baida & Gordijn, 2004). Moreover, given the -per case study- supplier-oriented service catalogue started with, in principle a significant amount of different bundles are possible (millions), which *serviguration* reduced by its reasoning process to a few relevant bundles (tenths), based on stated customer needs, and supplier-oriented relationships (and constraints) between services. So, serviguration is a good first attempt to arrive at automated configuration of a networked value constellation, in which a series of suppliers satisfy a need by bundling services. However, *serviguration* concentrates on conceptualizing services mainly from a *supplier* perspective and while it does have an ontology for taking customer needs into account, this needs ontology is only rudimentary. Most

importantly, the needs ontology from *serviguration* does not include the concept of a benefit, while this inclusion is important to differentiate between two apparently similar service offerings.

# 6 CONCLUSIONS

In this paper, we have shown how a catalogue of e-services can be created in a structured manner by applying the *e³service* ontology. Also, we have presented how to reason about finding e-service bundles to satisfy a specific need, using such a catalogue. In sum, the *e³service* ontology explicitly separates needs (problems) from wants (partial solutions) and demands (solutions of specific supplier). We then analyze the subjective economic value consequences for the customer (by consuming a service satisfying a want) to find additional wants, by following dependency relationships between want/consequence pairs. Thereafter, we select demands that can be provisioned by individual suppliers, based on the found wants and required benefits.

In future work, we will further elaborate upon these benefits, by showing that there is usually a mismatch between benefits as desired by the customer such as 'e-mail hosting' with 'web-based e-mail access' on the one hand, and the generic set of benefits as contained in a supplier-specific service such as 'e-mail hosting with web-based e-mail access', but also including 'virus scanning'. The research problem is then about making this mismatch as small as possible, by defining criteria that can be used to compare these mismatches to indeed select the smallest mismatch. Also, currently, we are working on software support for the *e³service* ontology.

**Acknowledgements.** We are grateful for the input of Pieter van der Ploeg from MalieNet, and Hans Burg and Frank Verlaan from Boer&Croon. This research has been partly funded by NWO/STW/Jacquard as the project VITAL.

## References

H. Akkermans, Z. Baida, and J. Gordijn. Value webs: Ontology-based bundling of real-world services. *IEEE Intelligent Systems*, 19(44):23—32, July 2004.

T. Andrews et al. Business process execution language for web services version 1.1, http://www.ebxml.org/specs/ebBPSS.pdf. Technical report, 2003.

J. Arndt. How broad should the marketing concept be? *Journal of Marketing*, 42(1):101–103, January 1978.

Z. Baida. *Software-aided service bundling*. PhD thesis, Free University Amsterdam, May 2006.

J. Gordijn and H. Akkermans. Value based requirements engineering: Exploring innovative e-commerce idea. In *Requirements Engineering Journal*, Vol. 8(2):114-134, 2003.

C. Grönroos. *Service Management and Marketing*. Lexington Books, 1990.

B. R. Group. The business motivation model - business governance in a volatile world. Technical report, 2005.

J. Gutman and T. Reynolds. Laddering theory-analysis and interpretation. *Journal of Advertising Research*, 28(1):11, Febraury/March 1988.

S. de Kinderen, J. Gordijn, *e³service*- an ontological approach for deriving multi-supplier e-service bundles from customer needs. In *Proceedings of the Forty-first Hawai'i International Conference on System Sciences (HICSS-41)*. Computer Society Press, jan 7-10 2007.

P. Kotler. *Marketing Management*. Prentice Hall, 2000.

C. Lovelock. *Service Marketing - People, Technology, Strategy*. Prentice Hall, 2001. Fourth edition.

J. Ramsay. The real meaning of value in trading relationships. *International Journal of Operations and Production Management*, 25(6):549–565, 2005.